

## REPORT DOCUMENTATION PAGE

AD-A210 016

2b. 1		1b. RESTRICTIVE MARKINGS	
		3. DISTRIBUTION / AVAILABILITY OF REPORT Approved for public release; distribution unlimited.	
4. PERFORMING ORGANIZATION REPORT NUMBER(S)		5. MONITORING ORGANIZATION REPORT NUMBER(S) <b>AFOSR-TR-89-0973</b>	
6a. NAME OF PERFORMING ORGANIZATION New York University	6b. OFFICE SYMBOL (If applicable)	7a. NAME OF MONITORING ORGANIZATION Air Force Office of Scientific Research/NL	
6c. ADDRESS (City, State, and ZIP Code) 550 1st Avenue New York, NY 10016		7b. ADDRESS (City, State, and ZIP Code) Building 410 Bolling AFB, DC 20332-6448	
8a. NAME OF FUNDING / SPONSORING ORGANIZATION AFOSR	8b. OFFICE SYMBOL (If applicable) NL	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER AFOSR-85-0341	
8c. ADDRESS (City, State, and ZIP Code) Building 410 Bolling AFB, DC 20332-6448		10. SOURCE OF FUNDING NUMBERS	
		PROGRAM ELEMENT NO. 61102F	PROJECT NO. 2313
		TASK NO. A5	WORK UNIT ACCESSION NO.
11. TITLE (Include Security Classification) Conformal Image Warping			
12. PERSONAL AUTHOR(S) Eric Schwartz			
13a. TYPE OF REPORT Reprint	13b. TIME COVERED FROM TO	14. DATE OF REPORT (Year, Month, Day)	15. PAGE COUNT 17
16. SUPPLEMENTARY NOTATION			
17. COSATI CODES		18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)	
FIELD	GROUP	SUB-GROUP	
19. ABSTRACT (Continue on reverse if necessary and identify by block number)			
<div style="text-align: right;"> <b>SDTI</b>  <b>ELEC</b>  <b>JUL 11 1989</b>  <b>H</b> </div>			
20. DISTRIBUTION / AVAILABILITY OF ABSTRACT <input checked="" type="checkbox"/> UNCLASSIFIED/UNLIMITED <input checked="" type="checkbox"/> SAME AS RPT <input type="checkbox"/> OTHER USES		21. ABSTRACT SECURITY CLASSIFICATION UNCLASSIFIED	
22a. NAME OF RESPONSIBLE INDIVIDUAL John F. Tangney		22b. TELEPHONE (Include Area Code) (202) 767-5021	22c. OFFICE SYMBOL NL

# Robotics Research Technical Report

AFOSR-EE-89-0873

*was*  
IEEE CGA (89)

# **Conformal Image Warping**

by

**Carl Frederick  
Eric L. Schwartz**

---

**Technical Report No. 408  
Robotics Report No. 174  
October, 1988**

**New York University  
Dept. of Computer Science  
Courant Institute of Mathematical Sciences  
251 Mercer Street  
New York, New York 10012**

**Work on this paper has been supported by AFOSR 85-0341, System Development Foundation, and the Nathan S. Kline Psychiatric Research Center.**

## CONFORMAL IMAGE WARPING

*Carl Frederick*

*Eric L. Schwartz*

Computational Neuroscience Laboratories  
New York University School of Medicine  
Department of Psychiatry  
550 First Avenue  
New York, N.Y. 10016

Robotics Research  
Department of Computer Science  
Courant Institute of Mathematical Sciences  
715 Broadway  
New York, N.Y. 10003

### ABSTRACT

This report describes numerical and computer graphic methods for conformal image mapping between two simply connected regions. The immediate motivation for this application is that the visual field is represented in the brain by mappings which are, at least approximately, conformal [4] [11] [16] [17]. Thus, in order to simulate the imaging properties of the human visual system (and perhaps other sensory systems), conformal image mapping is a necessary technique.

There are two distinct aspects to this problem: first, one must implement a numerical or analytic method which allows for the computation of a given conformal mapping, constrained by the shape of the two simply connected regions (hereafter known simply as regions) to be mapped, and by a single point and orientation correspondence between them; second, it is necessary to apply a space variant texture mapping algorithm to warp the image, once the mapping itself has been specified.

For generating the conformal map, we show a method for analytic mappings, and also an implementation of the Symm [13] algorithm for numerical conformal mapping. The first method evaluates the inverse mapping function at each pixel of the range, with anti-aliasing via multi-resolution texture pre-filtering and bilinear interpolation [18]. The second method is based on constructing a piecewise affine approximation of the mapping in the form of a joint triangulation, or triangulation map, in which only the nodes of the triangulation are conformally mapped. The texture is then mapped by a local affine transformation on each pixel of


the range triangulation with the same anti-aliasing as the first method.

We illustrate these algorithms with examples of conformal mappings constructed analytically from elementary mappings such as the linear fractional map, the complex logarithm, etc. We also show applications of numerically generated maps between highly irregular regions, and also an example of the visual field mapping which motivates this work.

In addition to providing a necessary tool for simulation of cortical architectures, these illustrations may be of pedagogical use for students who are attempting to visualize the geometric properties of elementary conformal mappings, and may also find application in areas such as fluid mechanics and electrostatics, where conformal mapping is a natural and basic tool.

October 14, 1988



Accession For	
NTIS GRA&I	<input checked="checked" type="checkbox"/>
DTIC TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification	
By	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
A-1	

# CONFORMAL IMAGE WARPING

*Carl Frederick*

*Eric L. Schwartz*

Computational Neuroscience Laboratories  
New York University School of Medicine  
Department of Psychiatry  
550 First Avenue  
New York, N.Y. 10016

Robotics Research  
Department of Computer Science  
Courant Institute of Mathematical Sciences  
715 Broadway  
New York, N.Y. 10003

## 1. Introduction: conformal mapping

Conformal mappings may be defined in a number of equivalent ways, which emphasize different aspects of their geometric or analytic properties [1]:

- Complex analytic functions  $f(z)$ , for  $\frac{df}{dz} \neq 0$ , represent conformal mappings.
- A conformal mapping is locally isotropic.  
This means that an infinitesimal area element is magnified equally in all directions.
- Infinitesimal angles are preserved by conformal mapping.
- The real and imaginary parts of the map function are harmonic conjugate functions, i.e. they satisfy the Laplace equation,  $\nabla^2 f(x,y) = 0$  and intersect orthogonally. This property provides important practical application to areas of potential theory (electrostatics, fluid mechanics, etc.) where the Laplace equation occurs.

The Riemann mapping theorem guarantees existence and uniqueness of conformal mappings between regions. Given a region, there exists a conformal mapping of this region onto the unit disk. The mapping is made unique by fixing the mapping of a single point in the region onto the center of the unit disk, and fixing the orientation of the unit disk (see appendix 1 for a discussion of the Riemann mapping theorem).

The problem of conformal mapping of textures (images) has two distinct parts: the map function itself must be provided, and since the scaling induced by the mapping can change continuously, texture mapping must be space variant. In the following, we will describe two different methods for implementing the conformal mapping (one analytic and one numerical) and two different methods of texture mapping (pixel-based and

polygon-based).

## 2. Elementary analytic maps

This method performs texture mapping by evaluating a given elementary analytic map on a pixel-by-pixel basis.

In the case of mappings such that the inverse or a desired branch of the inverse can be described analytically, a direct method is possible. A naive approach consists of evaluating  $f(z)$  at each point, copying the texture value at  $z$  to its image position in the  $w$ -plane. This method leads to significant aliasing in the  $w$ -plane. Instead we use the following method: We consider a small region of the  $w$ -plane; the inverse mapping  $f^{-1}(w)$  is determined and  $f^{-1}(w)$  is evaluated at each point within this region. The texture value at  $z = f^{-1}(w)$  is retrieved and used as the value at  $w$ . Effectively, we find all the points in the  $z$ -plane which map to our piece of the  $w$ -plane by  $f(z)$ . We still get aliasing by this method but it can be handled easily by one of the standard methods for pre-filtering [18] [3]. See figure 1 for examples of familiar elementary functions in complex variables. Figure 2 gives an example of a mapping generated by composition from fractional linear maps and elementary functions.

## 3. Numerical conformal mapping

The above example described a method for texture mapping elementary analytic mappings. In the present section, we describe a method for constructing numerical conformal mappings from one finite region to another which is applicable to the case where only the shapes of the two regions to be mapped (and a point and orientation correspondence) are known.

In this approach, one of the regions of interest is triangulated (e.g. using a Voronoi method [6]). The second region may then be triangulated by mapping the nodes of the initial triangulation to the second region, and then using the initial connectivity matrix† to triangulate the second region. It is important to point out, however, that this joint triangulation is potentially problematic. There is no guarantee that the connectivity matrix of a triangulation still describes a triangulation when its nodes have been mapped to a new region: some of its edges might then intersect, violating the definition of triangulation. For a given mesh, a joint triangulation can always be generated by adding pairs of points to the original point mapping, as proven by Saalfeld [10]. Saalfeld gives a constructive existence proof that the joint triangulation can be generated but this method requires exponential refinements (subdivisions) of its triangles. In our experience, no special means of generating valid triangle maps has been necessary as any reasonably well chosen mesh size leads to a valid triangle map.

We begin with the case of mapping an arbitrary region to the unit disk, and then discuss mapping between two arbitrarily shaped regions.

### 3.1. Mapping an arbitrary region to the unit disk

Henrici has recently surveyed existing numerical conformal mapping algorithms [8]. For arbitrary shaped regions (which are approximated by polygonal boundaries with large numbers of vertices), Symm's algorithm (Appendix 2) is preferable†.

†The connectivity matrix can be defined as a binary valued Cartesian product on the set of points in the region. If two points are connected by an edge, the matrix entry is 1, otherwise 0. In this way the topology of the triangulation map is specified.

†We thank Nick Trefethen for helpful discussions on this point.

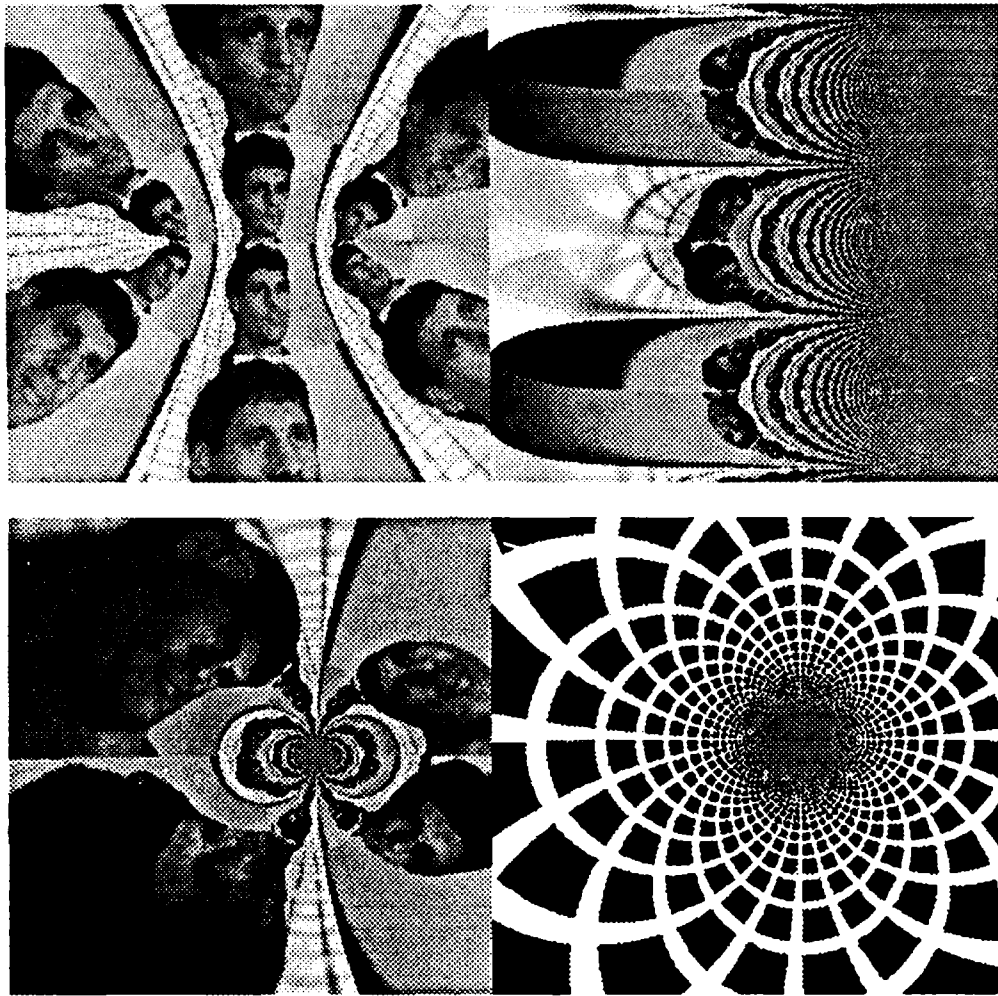


Figure 1.

Upper left: Complex cosine. The original texture (which has been repeated over the plane) is depicted in figure 2.

Upper right: Complex logarithm. Horizontal lines are images of lines radiating from the origin in the domain region. This image shows one complete "period strip" plus pieces of two others. Horizontal lines start to curve on the left due to floating point precision error.

Lower left and right: A fractional linear map,  $w = \frac{z-1}{z+1}$ , illustrated by the face image and also by a repeating rectangular lattice texture. The effect of this map on the plane can be imagined as poking a hole in the plane at the  $(-1,0)$ , turning it inside out and flipping it over.

The Symm algorithm is initialized with a description of the boundary of the region to be mapped, and the point in this region which is the pre-image of the center of the unit disk; it returns the mapping of the boundary and interior region to the unit disk.

One drawback of this method is that it provides the mapping of the desired region into the unit disk, but does not provide the inverse mapping.

The following is a description of the implementation of our algorithm. First, a point approximation of the mapping is constructed by applying Symm's interior mapping (Appendix 2) [13] to a sampling of the region at rectangular grid locations (the density of this grid is kept low for economy of computation and in many cases this type of



Figure 2.

This pair illustrates the conformal map which takes the unit disk to the upper half disk. The superimposed circle and half-circle isolate the regions in the planes which make this a useful transformation.

Its equation is: 
$$w = \frac{\left(\frac{1+z}{1-z}\right)^{\frac{1}{2}} - 1}{\left(\frac{1+z}{1-z}\right)^{\frac{1}{2}} + 1}$$

undersampling gives satisfactory results). The sampling is generated by a floating-point scan conversion algorithm operating as a polygon fill. The boundary points defining the region are mapped using Symm's boundary correspondence function. Second, this set of image points (interior and boundary) in the unit disk are triangulated using the Delaunay triangulation [9] and the same topology is applied to generate a triangulation of the points in the original region. See figure 3. Finally, texture mapping [2] is used to deform the texture in the region triangle by triangle to its unit disk counterpart. This is effectively a piecewise affine approximation of the conformal map. See figure 4.

### 3.2. Mapping one region to another region

To create a mapping from one region to another we start by generating the mapping from each region onto the unit disk, either analytically or using the Symm algorithm. We then triangulate the range set from the second region mapping. Now optimized point location [9] is used for each point in the range set of the first region mapping to determine which triangle from the range set of the second mapping contains that point. The affine mapping determined by this triangle and its counterpart in the domain of the second region mapping is used to map the included point from the first region range set to the second region. A cleaner solution would make use of a general method capable of mapping the unit disk to an arbitrary region, thus avoiding the numerical inversion of the Symm mapping. Some ideas on how this might be done are outlined by [8].

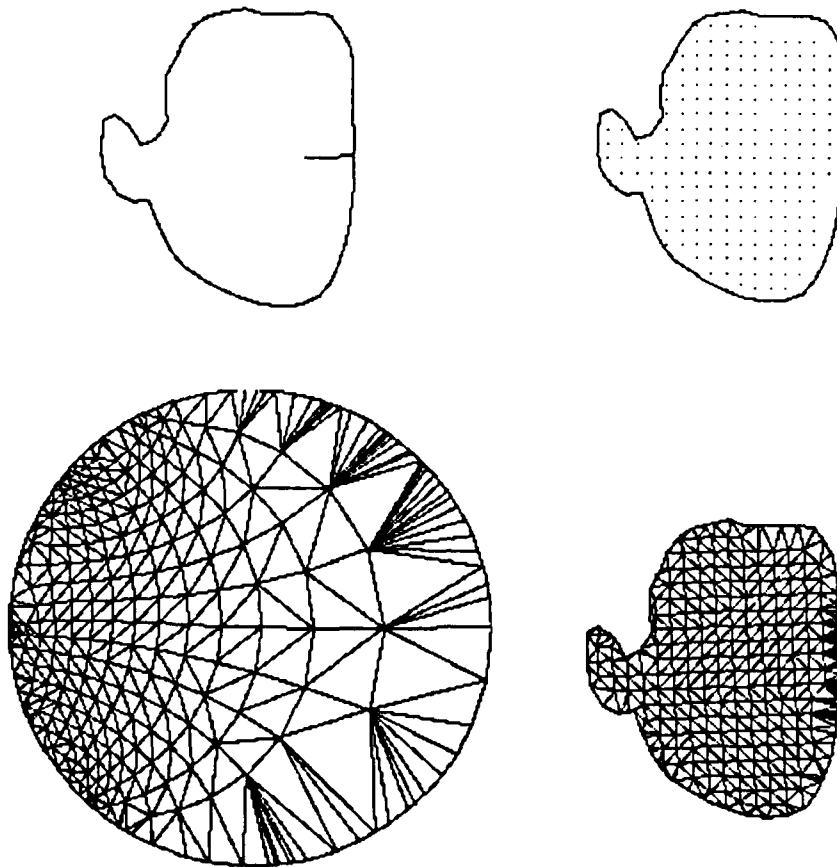


Figure 3.

Upper left: The original region is represented by a simple polygon. The vector pointing inward from the boundary indicates the uniqueness conditions imposed on the conformal mapping. Its interior end denotes the point which will be mapped to the origin of the unit disk. Its other end determines the rotation of the unit disk by specifying which point on the boundary will be mapped to the coordinates (1,0).

Upper right: The interior is partially filled with points on a rectangular grid using a floating point scan-conversion algorithm.

Lower left: The boundary and interior points are mapped to the unit disk using Symm's algorithm. This set of points is then triangulated.

Lower Right: The same triangulation connectivity information is used on the domain points to generate a domain triangulation.

#### ALGORITHM FOR MAPPING REGION TO REGION VIA THE UNIT DISK

**Px:** region boundary point set and pre-image of unit disk origin  
**Sx:** point set; either the sampled region or its disk mapping  
**Tx:** triangulation of Sx  
**Ax:** set of affine maps

```

(S1region1, S1disk) ← Symm(P1)
(S2region2, S2disk) ← Symm(P2)
T2disk ← Delaunay_triangulate(S2disk)
T2region2 ← connect S2region2 with same topology as T2disk
A2 ← generate_affine_maps_from(T2disk, T2region2)
For each point S1disk,i in S1disk
    j ← locate S1disk,i among T2disk
    S1region2,i ← A2j(S1disk,i)
T1disk ← Delaunay_triangulate(S1disk)
T1region1 ← connect S1region1 with same topology as T1disk
T1region2 ← connect S1region2 with same topology as T1disk
The joint triangulation is (T1region1, T1region2)

```

The above algorithm does not explicitly mention boundary points to simplify its presentation. To make the boundaries appear properly we do additional processing. Since this is a discrete approximation, points on the boundary of the first region are not likely to map to the points which define the boundary of the second region (unless the first boundary is very dense with points). The effect of mapping the boundary points from the first to the second region then appears to change the shape of the second region since the boundary points are mapped non-uniformly around the second region's boundary. In order to maintain the shape of the second boundary, its points are added to the points from the first region mapping. Thus, each boundary point from the disk-mapping of the second region is located between two of the boundary points from the disk-mapping of the first region. Using the affine transformation which maps the boundary edge in the first region disk-mapping back to the first region boundary edge, the point from the second region is mapped back to the boundary of the first region. All second region boundary points mapped to the first region in this way are then added to the original set from the first region. We call this augmented set of boundary points the refinement of the boundary. It is used as the new boundary of the mapping. In order to make the first region's boundary points map to the second region's boundary, the same method is used. This assures that the boundaries of both regions appear correctly. See figures 5 and 6.

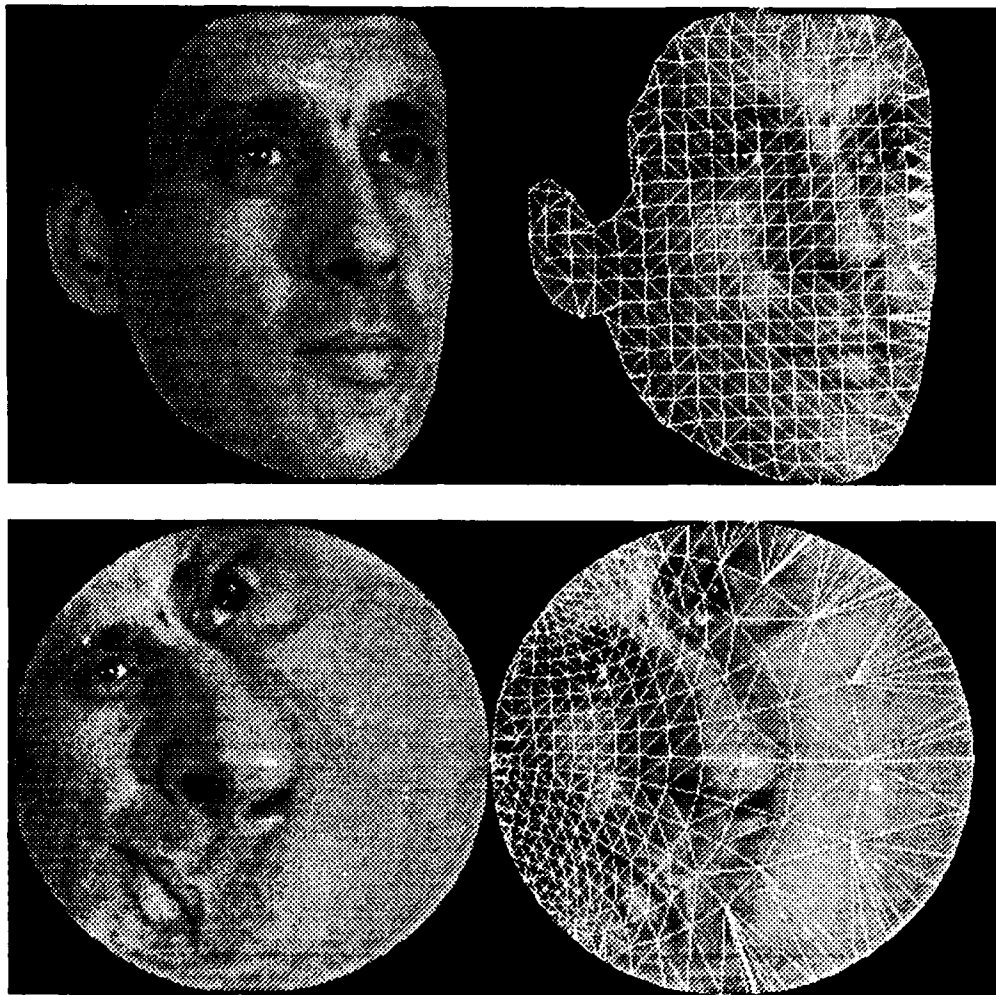


Figure 4.

Upper left: The original texture.

Upper right: Original texture overlaid with triangulation graph from figure 3.

Lower left: Face texture mapped onto unit disk.

Lower Right: Unit disk overlaid with triangulation graph from figure 3.

#### ALGORITHM FOR MAPPING REGION TO REGION VIA THE UNIT DISK WITH BOUNDARIES

```
(S1region1, S1disk) ← Symm(P1)
(S2region2, S2disk) ← Symm(P2)
T2disk ← Delaunay_triangulate(S2disk)
T2region2 ← connect S2region2 with same topology as T2disk
A2 ← generate_affine_maps_from(T2disk, T2region2)
A3 ← generate_boundary_affine_maps_from(S2disk, S2region2)
A4 ← generate_boundary_affine_maps_from(S1disk, S1region1)
For each point S1disk,i in the interior of S1disk
    j ← locate S1disk,i among T2region2
```

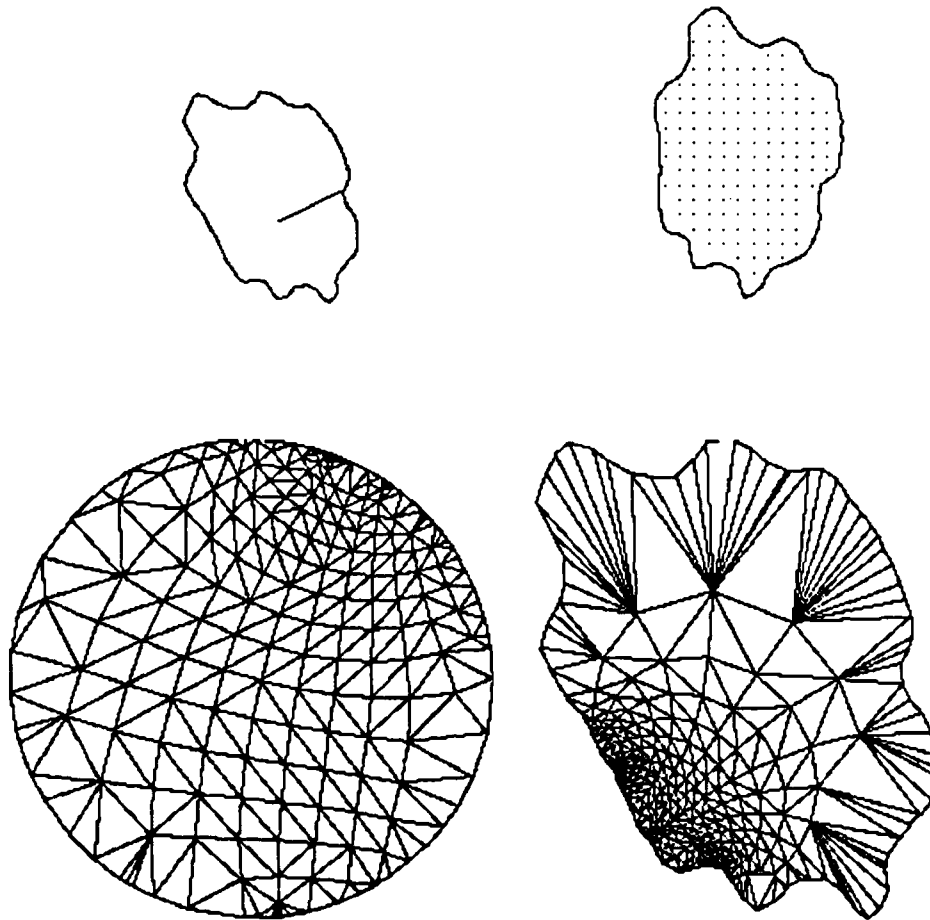
```

S1region2,i ← A2j (S1disk,i)
For each point S1disk,i on the boundary of S1disk
  j ← locate S1disk,i among boundary points of S2disk
  S1region2,i ← A3j (S1disk,i)
For each point S2disk,i on the boundary of S2disk
  j ← locate S2disk,i among boundary points of S1disk
  S2boundary1,i ← A4j (S2disk,i)
S1region1 ← {S1region1, S2boundary1}
S1disk ← {S1disk, boundary points of S2disk}
S1region2 ← {S1region2, boundary points of S2region2}
T1disk ← Delaunay_triangulate(S1disk)
T1region1 ← connect S1region1 with same topology as T1disk
T1region2 ← connect S1region2 with same topology as T1disk
The joint triangulation is (T1region1, T1region2)

```

#### 4. Application to visual cortex

In the case of visual cortex, there is considerable experimental evidence that the mapping of the retina to the surface of primary visual cortex is approximately isotropic. Thus, in order to model the representation of a visual image on the surface of the cortex, we need to construct a conformal approximation to this map, and to perform a conformal texture map of given visual field images. In order to illustrate this process, we show a numerical flattening that we have performed recently of the surface of primary visual cortex of the monkey [12]. In this work, we were able to identify a single point (the representation of the blind spot, or optic disk) in the eye, and an orientation (the orientation of the horizontal meridian). These observations, together with the flattened representation (and its boundary) were sufficient to generate the cortical map function [17]. The agreement of this methods of determining the cortical map, and direct micro-electrode measurements of the cortical map function, is excellent. Figure 7 shows a natural scene, mapped via this conformal approximation. The details of its construction are outlined in appendix 3.



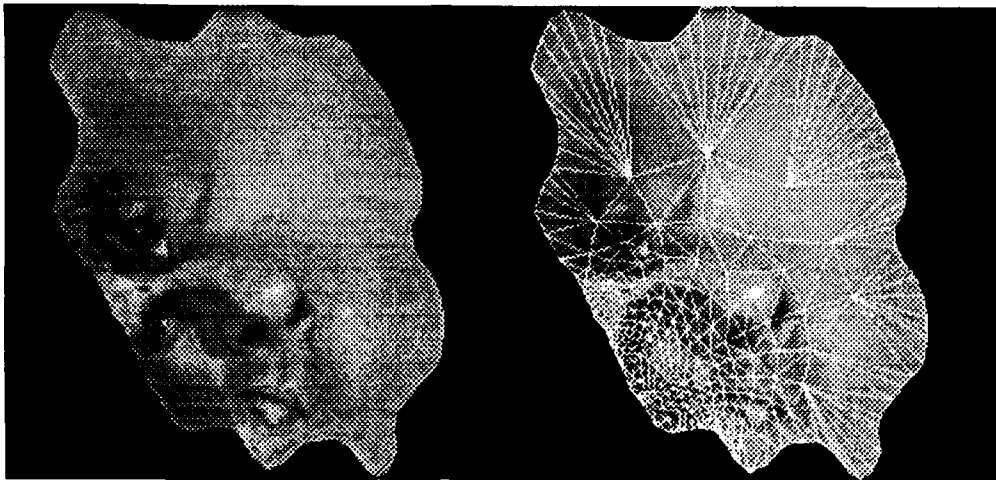
**Figure 5.**

These four are interpreted as in figure 3 except for the lower right corner which represents the range triangulation of the piecewise affine approximation of the conformal mapping of the region in figure 3 to the new region introduced here.

## **5. Comparison with earlier work**

An algorithm for the Schwarz-Christoffel method of generating conformal maps of polygonal regions has been developed by Trefethen [14]. Using Trefethen's algorithm as a basis, Fiume et al. [5] discuss interpolation of mesh approximations to such maps and the critical issue of filtering. However, this approach has little generality, since the conformal mapping algorithm which it uses is only adequate for polygonal approximations to regions which have no more than perhaps 10 sides [15]. For many actual applications of conformal mapping in computer graphic contexts, it is necessary to handle arbitrary domains, approximated by polygonal boundaries with large numbers of nodes. The method we present in the present paper is adequate for the general case, as suggested by some of the difficult domain shapes illustrated in the present paper.

Of the four methods described by Fiume for interpolation of a point-sampling of the mapping, the best is bilinear interpolation. Our approach is to use two-dimensional



**Figure 6.**

Here is the texture-mapping of the region specified in figure 3 to the region introduced in figure 5 via the unit disk as explained in the text. Note that the triangulation is the result of the piecewise affine approximation to the inverse of the conformal mapping which transformed the region in figure 5 to the unit disk. The highlight on the nose was chosen as the point to satisfy the first uniqueness condition of the Riemann mapping theorem. The orientation of the nose can be seen to follow the orientation vector in the upper left picture in figure 5.

linear interpolation over triangles as a first order approximation to the actual mapping. The geometric interpretation of linear interpolation can be imagined as finding the interpolated points as points on a triangulated polyhedral surface for each coordinate. This method is also known as barycentric coordinate interpolation, a fundamental building block of simple, stable interpolation within a computer graphics polygon rendering system and so of texture mapping polygons as well. Bilinear interpolation, geometrically interpreted, amounts to finding the interpolated points as points on a ruled surface where each surface patch of the ruled surface is defined by four mesh points. This ruled surface is smoother than the polyhedral surface but the difference between them is insignificant. For our purposes, bilinear or other higher order interpolation methods appear to be unnecessary and thwart the advantages of rendering in the comfortable world of triangles.

Once the conformal mapping has been approximated by a piecewise affine map, we use a standard pyramidal anti-aliasing technique [18] to filter the textured triangles. More sophisticated filtering techniques could be used, some of the most recent developments [7] involving space-variant kernels.

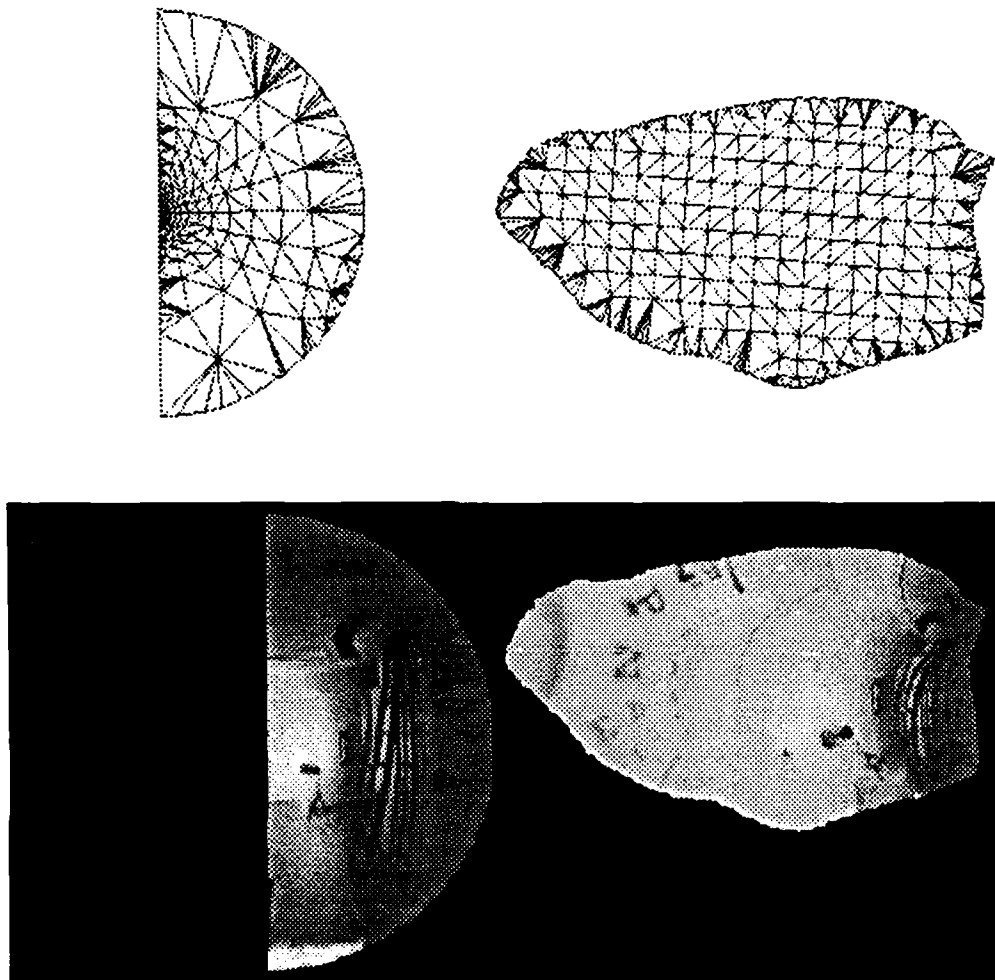


Figure 7.

The half-retina, flattened to a half-disk (on the left), is mapped to the flattened visual cortex. The fovea is at the midpoint of the vertical edge in the boundary of the retinal image. In cortex space, the area around the fovea is extremely enlarged due to the space-variant nature of the mapping. We used a special technique to obtain high resolution texture near the fovea which can not be represented at full resolution in a small picture. Thus, the area near the fovea in the retinal picture lacks detail.

## APPENDIX

### 1. Riemann mapping theorem

A typical statement of the Riemann mapping theorem is [1]:

Given any simply connected region  $\Omega$  which is not the whole plane, and a point  $z_0 \in \Omega$ , there exists a unique analytic function  $f(z)$  in  $\Omega$ , normalized by the condition  $f(z_0) = 0$ ,  $f'(z_0) > 0$ , such that  $f(z)$  defines a one-to-one mapping of  $\Omega$  onto the disk  $|w| < 1$ .

Our statement of this theorem is that uniqueness is specified by a point correspondence and an orientation. This is equivalent to the above statement: the point correspondence specifies the mapping of the point which maps into the origin of the unit circle

$f(z_0) = 0$ . The statement that  $f'(z_0) > 0$  is equivalent to fixing the orientation of the unit disk, since it specifies that only a (positive) scaling, and no rotation occur at this point. Other orientations of the mapping can be generated by multiplication of  $f(z)$  by  $e^{i\theta}$ .

## 2. Symm algorithm for conformal mapping

Symm has described an integral equation method for computing the conformal mapping of a given simply connected domain onto the interior of the unit circle. This method performs well for a domain described by a large number of (polygonal) vertices. It is based on the observation that the problem of conformally mapping a given simply connected domain  $D$  with boundary  $L$ , in the  $z$  plane, onto the unit disk  $|w| \leq 1$ , in the  $w$  plane, in such a way that a particular point  $z_0 \in D$  goes into the center  $w = 0$ , is given (up to arbitrary rotation) by:

$$w(z) = \exp[\log(z - z_0) + \gamma(z, z_0)]$$

where  $\gamma = g + ih$  and  $g$  satisfies the boundary conditions

$$\nabla^2 g = 0 \text{ for } z \in D,$$

$$g = -\log|z - z_0| \text{ for } z \in L,$$

This identity can be understood by observing that:

- $w(z_0) = 0$
- On the boundary ( $z \in L$ ),  $w(z) = e^{i(\arg(z - z_0) + h)}$ ,  
i.e.  $|w(z)| = 1$ , so the boundary  $L$  is mapped to the boundary of the unit circle.
- for interior points  $z \in D$ , we have  $|w(z)| \leq 1$  since  $|w(z)| = 1$  on the boundary, and  $w(z)$  must take its maximum in this region on the boundary, by the maximum principle [1].

Thus, Symm replaces the problem of finding  $w(z)$  to the problem of finding an analytic function  $\gamma(z) = \log(w) - \log(z - z_0)$ ; essentially, the problem is transformed from the original domain to a complex logarithmic representation of it.

Symm then goes on to outline methods of finding  $\gamma$ , by means of setting up a set of Fredholm integral equations, which are numerically solved by standard methods.

One difficulty in implementing this work, not emphasized in it, is that line integrals of the argument of an analytic function are evaluated in this algorithm. Direct (i.e. naive) numerical implementation of the equations found in this paper is not correct; careful attention must be paid to ensuring that a continuous branch of the argument is used when performing a line integral of the argument of a complex function.

## 3. Conformal Mapping of the Retinal Image to a Section of the Brain:

The problem of conformally mapping the retina to V1, e.g., can be regarded as a problem of mapping the half unit sphere (an approximation to the retina) conformally to a flattened model of the surface of visual cortex. For flattening of cortex, we use an algorithm developed in our lab which achieves an average error of roughly 5% in mapping a 3D representation of visual cortex into 2D. The mapping problem is solved in our case by decomposing it into three steps:

1. Mapping the quarter unit sphere (one half of a retina) conformally to the half unit disk.
2. Mapping the half unit disk conformally to the unit disk, where the point in the half disk which is mapped to the origin of the target disk is a parameter of this mapping, to be chosen dependent upon the specific data.
3. Mapping the unit disk conformally to an arbitrary 2-D domain, the flattened brain data.

**Step 1:**

We use the following stereographic projection (which provides a conformal mapping of the quarter unit sphere to the half unit disk):

$$(\theta, \phi) \rightarrow \left[ \frac{\cos\phi \sin\theta}{1 - \cos\phi \cos\theta}, \frac{\sin\phi}{1 - \cos\phi \cos\theta} \right],$$

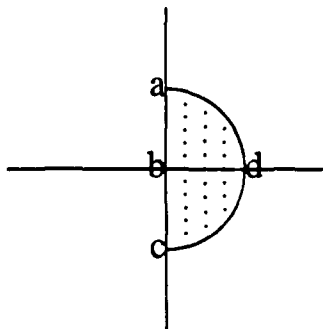
where  $\theta$  and  $\phi$  are the corresponding eccentricity and azimuth of the unit sphere, and the output represents 2D Cartesian coordinates.

**Step 2:**

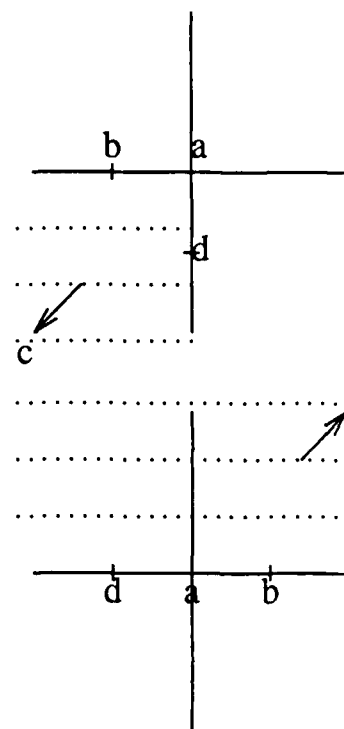
Let a point  $z = (x, y)$  on the plane represent a complex number. Then, a mapping of the half unit disk to the unit disk can be regarded as a complex transformation from the complex plane to itself. The following transformation  $T$  is a conformal complex transformation which maps the half unit disk to the unit disk, where a point  $\omega$  internal to the half unit disk is mapped to the origin of the target unit disk.

$$T = H \circ G \circ F; \quad T, F, G, H : \mathbb{C} \rightarrow \mathbb{C}.$$

Where:

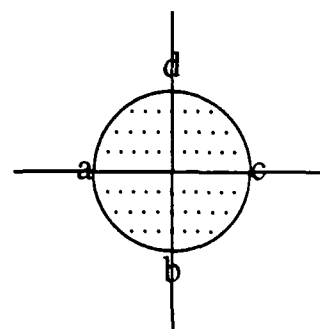


$$F(z) = \frac{z-i}{z+i} \rightarrow$$



$$G(z) = z^2 \rightarrow$$

$$H(z) = e^{i\alpha \frac{z-z_0}{z-\bar{z}_0}} \rightarrow$$



**Step 3:**

We use Symm's algorithm to conformally map a given simply-connected domain onto the interior of the unit circle. The results of this approximation have been checked against direct micro-electrode measurement of the map function of visual cortex [17], with excellent agreement.

## References

1. Ahlfors, L., in *Complex Analysis*, McGraw-Hill, New York, 1966.
2. Blinn, J. F. and M. E. Newell, Texture and Reflection in Computer Generated Images, *Communications of the ACM* 19, (October 1976), .
3. Crow, F., Summed area tables for texture mapping, *Computer Graphics (SIGGRAPH proceedings)* 11, 3 (1984), 200-220.
4. Daniel, D. M. and D. Whitteridge, The representation of the visual field on the cerebral cortex, *J. Physiology* 159, (1961), 203-221.
5. Fiume, E., A. Fournier and V. Canale, Conformal Texture Mapping, *Eurographics proceedings*, , Summer 1987.
6. Fortune, S., A sweepline algorithm for Voronoi diagrams, *ACM Symposium on Computational Geometry*, , 1987.
7. Fournier, A. and E. Fiume, Constant-Time Filtering with Space-Variant Kernels, *Computer Graphics (SIGGRAPH proceedings)* 22, 3 (August 1988), .
8. Henrici, P., in *Applied and computational complex analysis*, vol. 3, Wiley, New York, NY, 1986, 402-413.
9. Preparata, F. P. and M. I. Shamos, *Computational Geometry: An Introduction*, Springer-Verlag, New York, 1985.
10. Saalfeld, A., Joint triangulations and triangulation maps, *ACM symposium on computational geometry*, , 1987, 195-204.
11. Schwartz, E. L., On the mathematical structure of the retinotopic mapping of primate striate cortex, *Science* 227, (1985), 1066.
12. Schwartz, E. L., B. Merker, E. Wolfson and A. Shaw, Computational neuroscience: Applications of computer graphics and image processing to two and three dimensional modeling of the functional architecture of visual cortex, *IEEE Computer Graphics and Applications*, , (In press) 1988c.
13. Symm, G. T., An integral equation method in conformal mapping, *Numerische Mathematik* 9, (1966), 250-258.
14. Trefethen, L., Numerical computation of the Schwarz-Christoffel transformation, *SIAM Journal of Scientific and Statistical Computing* 1, (1980), 82-102.
15. Trefethen, L., private communication, , 1986.
16. Van Essen, D. C., W. T. Newsome and J. H. R. Maunsell, The visual representation in striate cortex of the macaque monkey: Assymetries, anisotropies, and individual variability, *Vision Research* 24, (1984), 429-448.
17. Weinshall, D. and E. L. Schwartz, A new method for measuring the visuotopic map function of striate cortex: validation with macaque data and possible extension to measurement of the human map, *Neuroscience Abstracts*, Washington, D.C., November 1987, 1291.
18. Williams, L., Pyramidal Parametrics, *ACM computer graphics, (SIGGRAPH proceedings)* 10(3), (1983), 1-30.

## **ACKNOWLEDGMENT**

We thank Nick Trefethen for helpful discussions, and also Adi Perry and Daphna Weinshall for contributing to early stages of this work.